



# **Relatório Técnico**

**Núcleo de  
Computação Eletrônica**

**Um Tutorial Inteligente  
para o Ensino de Estruturas  
de Dados e seus Algoritmos**

**C. Arisa  
C. Lima**

**NCE -18/02**

**Universidade Federal do Rio de Janeiro**

# **UM TUTORIAL INTELIGENTE PARA O ENSINO DE ESTRUTURAS DE DADOS E SEUS ALGORITMOS**

**Comissão Técnica: Informática**

**Cíntia ARISA**

**Universidade Federal do Rio de Janeiro  
Instituto de Matemática/Núcleo de Computação Científica**

**Universidade do Estado do Rio de Janeiro**

**Diretoria de Informática**

[cintia@posgrad.nce.ufri.br](mailto:cintia@posgrad.nce.ufri.br)

**Cabral LIMA**

**Universidade Federal do Rio de Janeiro**

**Departamento de Ciência da Computação/Instituto de Matemática**

[clima@dcc.ufri.br](mailto:clima@dcc.ufri.br)

## **RESUMO**

O aprendizado de um *savoir-faire* pode ser considerado uma atividade não trivial. Em processos de educação a distância existe uma complexidade adicional devido à dicotomia tempo/espço permeada nas atividades de ensino e aprendizagem, adicionada à necessidade de utilização apropriada das novas tecnologias.

As ferramentas tecnológicas estão usualmente agrupadas em plataformas, normalmente projetadas e implementadas para atender a alguns requisitos básicos (disponibilizar material instrucional, comunicar síncrona e assincronamente etc). Infelizmente, em projetos desse tipo de plataforma, os aspectos tecnológicos têm recebido mais atenção do que a própria aquisição de conhecimentos ou o uso de estratégias de ensino.

Neste trabalho, descrevemos um sistema tutorial inteligente, destinado ao ensino à distância de estruturas de dados e seus algoritmos correlacionados. A compreensão dos estudantes nessa área da Ciência da Computação depende de uma integração fina entre visualização dinâmica e raciocínio lógico. O sistema objetiva prover facilidades para a aprendizagem, via uso dinâmico de operadores contextuais, utilizando técnicas oriundas da inteligência computacional e do paradigma de orientação a objetos. Ressaltamos também a possibilidade de utilização em cursos presenciais, visando a liberação do professor das tarefas repetitivas de ilustração gráfica.

**Palavras-chaves:** Educação a Distância; Sistemas Tutoriais Inteligentes; Estruturas de Dados.

## **1. INTRODUÇÃO**

Nos dias de hoje, a competitividade faz com que profissionais das mais diversas áreas vivam em uma busca constante de conhecimento e aperfeiçoamento, muitas

vezes dificultada devido principalmente às suas limitações de horário e deslocamento. Nesse contexto, a educação à distância surge como uma tentativa de flexibilizar e democratizar o acesso ao ensino, tentando eliminar as barreiras de tempo e distância.

O ensino por si só já é uma atividade complexa e, na EAD, mais especificamente via Internet, podemos adicionar outras dificuldades, inerentes a essa modalidade. Segundo Lígia Leite em (Leite, 2000) a educação remota exige o desenvolvimento de um modelo pedagógico específico. Esse modelo pedagógico deve aproveitar o atual nível de desenvolvimento tecnológico na área de multimídia e Internet no sentido de motivar os alunos e maximizar o grau de aprendizado. A autora ainda afirma:

*“O modelo desenvolvido para a EAD baseada em material impresso não parece adequado ao ambiente online, não aproveitando os melhores recursos que este ambiente oferece”.* (Leite, 2000)

Apresentamos neste artigo nossa pesquisa, que consiste no projeto e implementação de um sistema destinado ao ensino à distância de tópicos relacionados a estruturas de dados e seus algoritmos. Esta ferramenta deverá ser integrada a uma plataforma de educação à distância de ampla abrangência (dedicada ao ensino remoto de diversos eixos pragmáticos de Ciência da Computação) atualmente sendo desenvolvida por nosso grupo de pesquisa.

## 2. MOTIVAÇÃO

É sabido que existe uma grande dificuldade no ensino clássico de estruturas de dados pois, de um modo geral, os alunos não dispõem de facilidades para o aprendizado efetivo do funcionamento dessas estruturas. Essa limitação é, sobretudo, concernente ao dinamismo visual necessário à compreensão do funcionamento lógico de estruturas tais como pilhas, filas e árvores (bem como a situação da memória durante a utilização das mesmas) além do fato delas serem geralmente implementadas através de ponteiros (Szwarcfiter e Markenzon, 1994), um conceito notadamente de entendimento não trivial.

A compreensão estática do aluno pode limitar o seu processo de aprendizagem e gerar lacunas que eventualmente irão produzir um efeito em cascata para outras subáreas da Ciência da Computação. Deficiências no aprendizado de estruturas de dados podem ser fatores decisivos para a não compreensão de banco de dados, por exemplo.

A figura 1 exemplifica uma possibilidade de representação lógica de uma estrutura do tipo lista linear simplesmente encadeada.

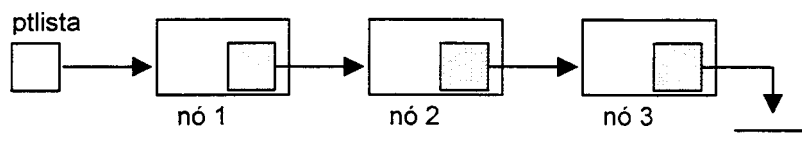


Fig. 1 – Lista linear simplesmente encadeada

Cada nó (elemento) da lista armazena suas informações e possui um campo do tipo ponteiro, que faz referência ao próximo nó, ou seja, contém o endereço de memória

do elemento subsequente na estrutura lógica. Uma inclusão ou exclusão de elemento na lista implica em alteração de vários ponteiros, de forma que a lista não sejam perdidas as referências aos nós subsequentes. Essa manipulação de ponteiros é ainda dificultada pelo fato de que os nós da lista, na maioria das vezes, são implementados através de variáveis dinâmicas e portanto anônimas, só sendo possível acessá-los através de operações de indireção sobre as variáveis ponteiro.

A figura 2 mostra respectivamente trechos de algoritmos que buscam um elemento em uma lista encadeada e efetuam a exclusão do elemento a partir de seu endereço (Szwarcfiter e Markenzon, 1994). O entendimento do algoritmo, repleto de ponteiros e indireções, costuma ser uma atividade complexa para a maioria dos estudantes.

Busca em uma lista ordenada	Remoção do nó apontado por pont
<pre> Procedimento busca-enc (x,ant,pont)   ant := ptlista;  pont := λ   ptr := ptlista↑. Prox   enquanto ptr ≠ λ faça     se ptr↑.chave &lt; x então       ant := ptr       ptr := ptr↑.prox     senão se ptr↑.chave = x então       pont := ptr       ptr := λ </pre>	<pre> busca-enc (x,ant,pont) se pont ≠ λ então   ant ↑.prox := pont ↑.prox   valor recuperado := pont ↑.info   desocupar (pont) senão "nó não se encontra na tabela" </pre>

Fig. 2 – Algoritmos de manipulação de listas encadeadas

A despeito das dificuldades inerentes à complexidade do assunto, é mister que se diga que alguns alunos conseguem atingir um nível de aprendizado satisfatório. Isso possivelmente ocorre, graças a estratégias adotadas por determinados professores, que usualmente utilizam representações gráficas a fim de possibilitar uma visualização mais eficaz do funcionamento das estruturas lógicas dos dados e de suas respectivas representações em memória, atuando eles próprios como simuladores. Entretanto esse dinamismo ocorre apenas enquanto a aula é ministrada, pois mesmo considerando uma boa capacidade técnica de representação gráfica por parte do professor presencial, habitualmente, após a explicação devidamente anotada em seu material, os alunos ficam de posse de uma representação estática e frequentemente inapropriada para um estudo a posteriori.

Após a exclusão do nó 2 em nossa lista encadeada, por exemplo, teríamos a representação gráfica correspondente a figura 3, que não se mostra muito informativa com relação às alterações executadas na estrutura, possibilitando que o aluno, ao consultar a figura, permaneça com dúvidas a respeito da maneira como a representação foi atingida. Isso acontece porque apenas as estruturas resultantes de operações são transcritas, e não a representação do funcionamento e modificações dessas estruturas até a obtenção dos resultados finais. Para o estudo futuro, os livros e cadernos estarão armazenando a informação estática, fazendo uso, no máximo, de recursos gráficos tais como setas pontilhadas para representar novas referências e "X"

para representar os elementos removidos (figura 4), correndo até mesmo o risco de poluir visualmente a representação.

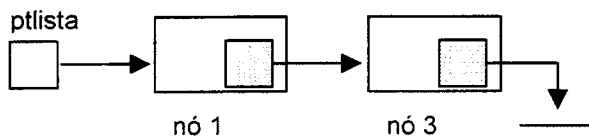


Fig. 3 – Lista após a exclusão do nó 2

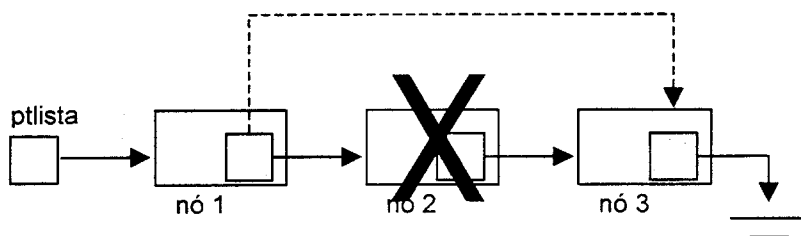


Fig. 4 – Representação das alterações decorrentes da remoção do nó 2

Posto que, a princípio, apenas o professor domina com relativa perfeição a técnica de construção e manipulação das estruturas e seus algoritmos, faz-se necessário um tempo bastante expressivo de estudo em conjunto (professor e aluno) para que os estudantes possam praticar e testar a sua aprendizagem.

O grau de entendimento dos alunos poderia ser maximizado se houvesse uma maneira de armazenar o comportamento dinâmico da estrutura, ou seja, todas as alterações sofridas, na ordem em que ocorreram e com referência à linha de execução correspondente no algoritmo.

É de fácil percepção que em processos de educação a distância, temos que considerar alguma complexidade adicional relativa às características peculiares de espaço e tempo. Nesta modalidade de ensino, o aluno normalmente estuda sozinho, utilizando o material disponibilizado remotamente pelo professor/tutor. Esse material na maioria das vezes é estático e portanto não adequado para expressar a dinamicidade inerente às estruturas de dados.

Além disso, mesmo que consiga ser entendida, tal representação gráfica demonstra apenas uma possibilidade de exclusão. O aluno pode sentir necessidade de testar outras, como: se o nó a ser excluído for o primeiro da lista? E se for o último? E se a lista possuir apenas um elemento? E se a lista estiver vazia? Muitos desses casos são exceções, o que significa que o comportamento da estrutura não será necessariamente o mesmo. Em sala, o professor poderia simular cada uma dessas situações, mas o tempo gasto com essa atividade seria, muitas vezes, proibitivo. A cada nova necessidade de entendimento de alguma manipulação da lista, o professor precisaria ser consultado e teria que dispendir o tempo necessário para simular a operação. Essa atividade seria significativamente dispendiosa, em termos de tempo, para o ensino presencial. Além do mais, a presença de representações gráficas para cada caso possível de manipulação de cada tipo de estrutura em livros, cadernos e apostilas é praticamente inviável.

Esse problema mostra-se ainda mais acentuado devido às dificuldades de *feedback* da Educação a Distância, quando os estudantes normalmente não dispõem da participação imediata do professor para dirimir dúvidas pontuais sobre ilustrações e/ou execuções de algoritmos dedicados. É de se ressaltar que apesar da possibilidade de interação remota entre aluno/tutor, na prática, ela dificilmente acontece em tempo real. Em outras palavras, usualmente neste processo de aprendizagem as simulações dinâmicas, tão necessárias para o sucesso do ensino das estruturas e seus algoritmos, ficam relegadas a algumas tarefas específicas e de periodicidade discreta. Não é comum, em plataformas de EAD, a possibilidade de simulações síncronas abordando estruturas de dados e algoritmos correlatos. As alterações não monotônicas, relativas a esta área de conhecimento, deveriam ser objeto de visualizações dinâmicas das estruturas, permitindo, entre outras coisas, a compreensão exata dos diversos estágios pelos quais uma estrutura passa antes de atingir um estado final de representação.

Outro agravante é que estruturas de dados não são sempre tão simples quanto a exemplificada nesse artigo. Quando nos referimos a estruturas do tipo árvore, por exemplo, a inclusão ou exclusão de um elemento pode ser ainda mais complexa, implicando em diversas operações como rotações, cisões, concatenações, redistribuições e outras. Como ilustração, podemos reparar na árvore binária completa<sup>1</sup> representada inicialmente na figura 5.

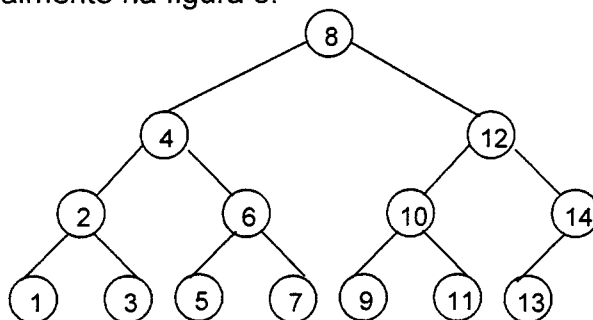


Fig. 5 – Árvore binária completa

Após a inserção do nó 0, todos os ponteiros precisam ser alterados, tornando a nova árvore da figura 6 bastante diferente da anterior. A representação gráfica em material de todas as alterações sofridas até a obtenção do resultado final é bastante complexa, sendo praticamente inviável.

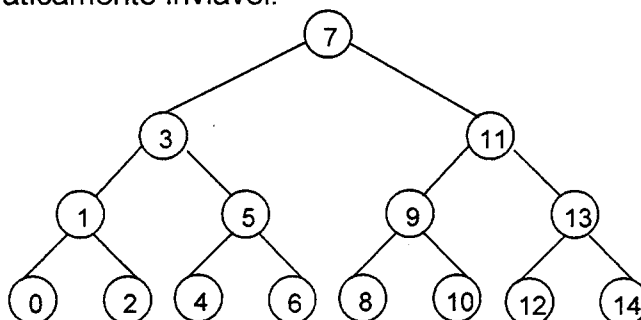


Fig. 6 – Árvore binária completa após a inserção do nó 0

<sup>1</sup> Em uma árvore binária completa, qualquer nó que possua subárvore vazia, se encontra no penúltimo ou último nível da árvore (Szwarcfiter e Markenzon, 1994).

### 3. PROPOSTA

Neste trabalho propomos o projeto e a implementação de um sistema dedicado ao ensino/aprendizagem de estruturas de dados e algoritmos correlacionados. Este sistema, orientado a objetos, sendo implementado em Java, utilizaria técnicas advindas da inteligência computacional e seria suficientemente parametrizável, visando o acoplamento dinâmico a uma plataforma de EAD. A abstração imposta ao sistema, o autoriza a usufruir de eventuais diagnósticos uniformes em elaboração síncrona e flutuantes na plataforma a fim de adaptar-se ao contexto dos processos de ensino e aprendizagem (LIMA, 2000).

Através desse sistema tutorial com simulador interativo, os alunos poderão criar listas, pilhas, filas e árvores e verificar as modificações em suas estruturas lógicas, bem como em suas organizações internas de memória, ao longo de diversos tipos de manipulação de seus elementos. O aprendiz poderia executar exaustivas repetições no momento desejado, sem a dependência da presença constante do professor para representar graficamente as alterações efetuadas.

O sistema, ilustrado na figura 7, será capaz de exibir o estado inicial da estrutura e, ao longo das modificações impostas, todas as alterações dinamicamente sofridas por ela, estando habilitado portanto a atender a uma determinada solicitação desde o início, até a obtenção de seu estado final.

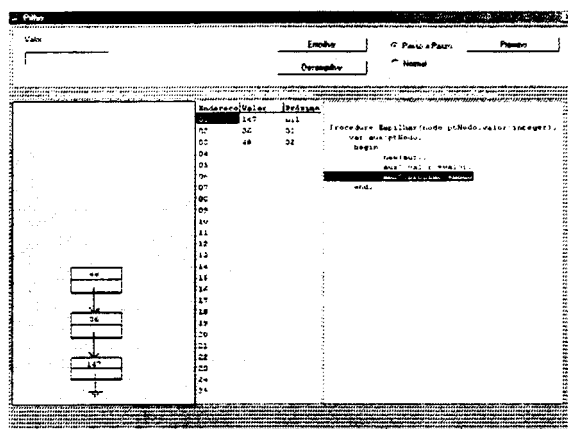


Fig. 7 - Tela de simulação de pilha

A interatividade permite que o aluno possa decidir qual operação deseja realizar a cada momento, bem como sobre qual elemento de qual estrutura.

O aluno poderá verificar os algoritmos utilizados, dedicados às operações na estrutura e também teria a possibilidade de executá-los passo a passo, com uma visualização gerada por um subsistema de *trace*. A cada operação seria ilustrada a instrução que motivou a alteração bem como seu efeito sobre a representação lógica da estrutura.

Um sistema dedicado aos processos de ensino/aprendizagem é um meio bastante adequado para o armazenamento de conhecimentos dinâmicos que regem o funcionamento das estruturas de dados. Os livros, as apostilas ou mesmo as anotações feitas em sala de aula são, naturalmente, repositórios estáticos que frequentemente não

representam com fidedignidade as modificações sofridas pela estrutura no momento de operações a ela impostas. Utilizando o sistema, o aluno poderá aplicar as operações sobre essas estruturas quantas vezes tiver necessidade e, além de verificar “o que” aconteceu, saberá também “como” aconteceu.

Por exemplo, na remoção do segundo nó da lista simplesmente encadeada da figura 2, teríamos a seguinte sequência de imagens geradas pelo sistema (figura 8)

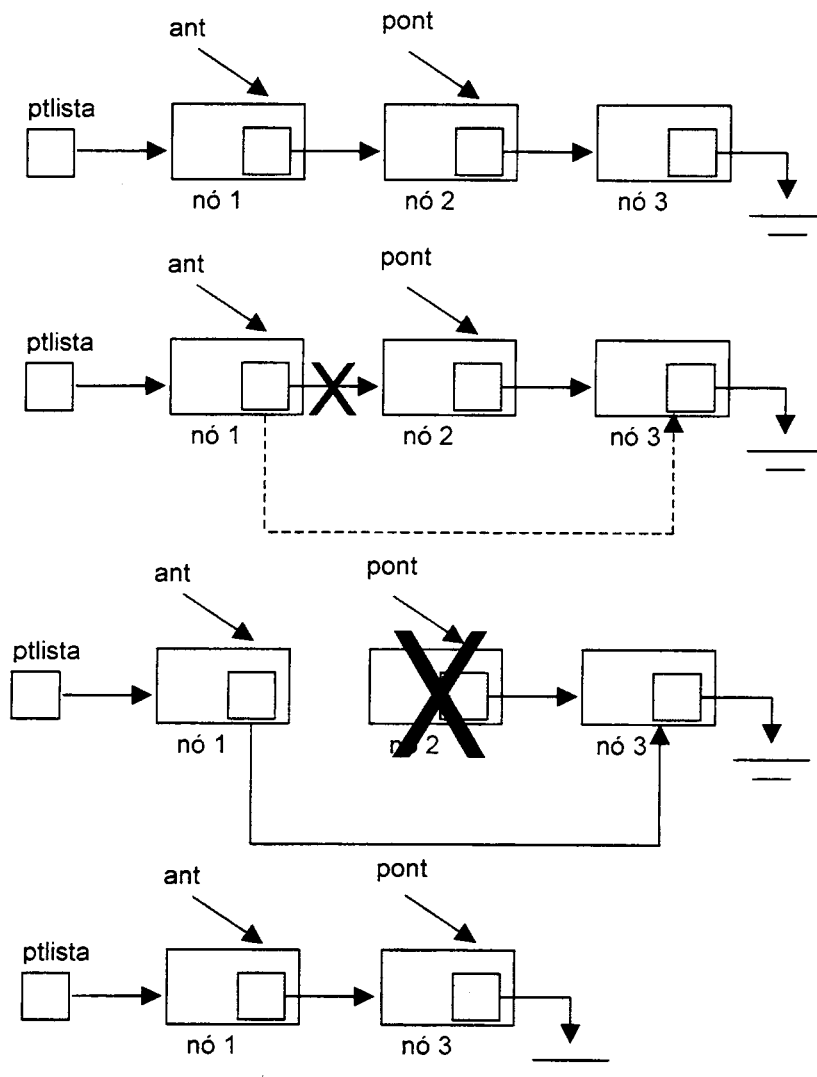


Fig. 6 – Eliminação do nó 2, passo a passo

Além da visualização do funcionamento dos algoritmos pré-definidos, quando da execução de exercícios, o aluno pode ter necessidade de criar seus próprios algoritmos de manipulação das estruturas e testá-los, identificando, por exemplo, a ocorrência de erros de lógica. O sistema será capaz de prover a visualização da execução também para os algoritmos desenvolvidos pelos próprios alunos. No entanto, a alteração incorreta de apenas um único ponteiro pode por tudo a perder, fazendo com que o



algoritmo não funcione da maneira esperada. Com o objetivo de auxiliar o aluno na árdua tarefa de identificação dos erros, o sistema contará com um mecanismo de help intrusivo. Dessa forma, as atividades do aluno no sistema serão monitoradas na tentativa de identificar as possíveis motivações para a ocorrência do erro. O aluno será automaticamente alertado quanto a prováveis deficiências teóricas que podem estar ocasionando o erro e serão indicados assuntos a serem revistos bem como alguns caminhos para a solução do problema. O sistema deverá incluir também um mecanismo tradicional de ajuda interativo, onde o estudante poderá dirimir dúvidas conceituais a respeito de tópicos relacionados às estruturas de dados.

A despeito de ser uma ferramenta originalmente dedicada aos processos de ensino/aprendizagem remotos, este sistema poderia também ser normalmente utilizado na modalidade presencial, pois apesar do professor lançar mão em sala de aula de abstrações de alto nível, ele precisaria executar um número exaustivo de repetições até que os alunos pudessem atingir um nível de compreensão adequado e satisfatório. Essa atividade repetitiva de simulação do funcionamento das estruturas e suas reações a alterações impostas poderá ser executada pelo sistema, liberando portanto o professor presencial para outras atividades mais nobres, onde sua atuação seja realmente indispensável. Por consequência direta dessa utilização em termos presenciais, vislumbra-se, minimamente, uma possibilidade de ajuda na representação mental do aluno das diversas estruturas de dados e seus algoritmos, podendo ser de grande importância na assimilação do conteúdo ministrado em aula.

Este sistema poderá ser utilizado de forma pontual ou acoplado a plataformas de ensino-aprendizagem. Em nossa proposta o sistema deverá estar acoplado à plataforma ACÁDIA, um ambiente projetado e desenvolvido de forma a estimular a colaboração e fomentar a criatividade de alunos e professores, colocando-os em papel ativo no desenvolvimento de sua aprendizagem (BRASIL e MARTINS, 2002). Essa plataforma deverá abrigar diversas ferramentas, destinadas ao ensino dos mais variados tópicos da área de Ciência da Computação, como Design Patterns (LIMA e GANDRA, 2002) e Orientação a Objetos (LIMA et al, 2002), bem como outras áreas de conhecimento.

#### **4. CONCLUSÃO**

É praticamente um consenso entre os educadores a complexidade inerente ao ensino de estruturas de dados e algoritmos correlacionados. Essas dificuldades normalmente são minimizadas quando o aluno consegue visualizar o funcionamento lógico da estrutura no momento da execução das operações de manipulação. A tarefa de representação gráfica é geralmente executada com relativo sucesso pelos professores em sala de aula, mas como auxiliar o estudo posterior e ainda, o que fazer a respeito da educação à distância?

Os materiais didáticos disponibilizados nos cursos presenciais e à distância consultados pelos alunos são, na sua maioria, estáticos, não sendo adequados para representar o dinamismo visual tão necessário nessa área de conhecimento.

Nossa proposta consiste na utilização de um sistema tutorial inteligente, com o objetivo de minimizar as dificuldades de aprendizado, funcionando como um repositório mais adequado para informações dinâmicas.

Trata-se de um sistema interativo, de tal forma que o aluno possa simular as mais diversas operações, não estando restrito apenas àquelas disponibilizadas como exemplos.

O sistema deverá utilizar recursos baseados em inteligência computacional com o objetivo de permitir que os aprendizes criem seus próprios algoritmos de manipulação de estruturas e possam se certificar de sua correção. Além disso, um mecanismo de *help* intrusivo seria capaz de monitorar as atividades dos estudantes durante a utilização do tutorial, fornecendo indicações a respeito das prováveis deficiências teóricas e sugerindo tópicos para estudo bem como atividades de revisão e fixação dos conteúdos.

Apesar de originalmente idealizado para cursos à distância (podendo ser acoplado a plataformas de EAD), onde a separação espaço-temporal e as restrições de feedback podem acentuar ainda mais as dificuldades de aprendizado, o sistema também oferece a possibilidade de ser utilizado como ferramenta de apoio em cursos presenciais. Dessa forma os estudantes terão um auxílio substancial fora do ambiente de sala de aula, beneficiando a assimilação de conteúdo e também evitando a dependência excessiva do professor para simulação de todas as operações possíveis sobre as estruturas de dados.

## 5. REFERÊNCIAS BIBLIOGRÁFICAS

Brasil, Bárbara e Martins, Cristiana – **Acádia - Ambiente de ensino-aprendizagem com ênfase em comunicação gráfica, desenvolvido em plataforma distribuída** - Núcleo de Computação Eletrônica da Universidade Federal do Rio de Janeiro – Brasil - 2002.

Chaves, Eduardo - **Ensino a Distância – Conceitos Básicos**, 1999 - Disponível na Internet via <http://www.edutecnet.com.br/Tecnologia%20e%20Educacao/edconc.htm>. Arquivo consultado em outubro de 2001.

Leite, Lígia S. - **Temas e questões estratégicas para o desenvolvimento da EAD**, 2000 - Disponível na Internet via [http://www.intelecto.net/desenvolvimento\\_ead.htm](http://www.intelecto.net/desenvolvimento_ead.htm). Arquivo consultado em abril de 2002.

Lima, Cabral – **Software packages for Distance Learning Purposes**. Research Report 01/2000, CCTUENF, Brasil, 2000

Lima, Cabral e Gandra, Henrique - **LeSOOP: a System Aimed to the Remote Learning of Design Patterns in Object-Oriented Systems – Proceedings of International Symposium on System Integration (INTERSYMP2002)**, Baden-Baden, Alemanha, 2002 (em edição)

Lima, Cabral et al. - A Remote Learning Environment for Object-Oriented Technologies.(TOOHELP) - **Proceedings of International Symposium on System Integration (INTERSYMP2002)**, Baden-Baden, Alemanha, 2002 (em edição)

Szwarcfiter, Jaime L. e Markenzon, Lilian – **Estruturas de Dados e seus Algoritmos.**, Rio de Janeiro, LTC Editora, 1994, 320p